A modular approach to Integrated Assessment modeling

Marian Leimbach*
Carlo Jaeger
PIK - Potsdam Institute for Climate Impact Research,
P.O. Box 60 12 03, D-14412 Potsdam, Germany

Abstract

In this paper, we present a new approach to model coupling that probably forms the methodological basis of a new generation of Integrated Assessment models. This approach respects the knowledge and expertise that is embodied in existing models and encourages their gradual evolution. Modularity is the guiding principle. Our approach is distinguished by the way modules are coupled which is based on an interplay of a job control module, a numerical coupling module, and a couple of stand-alone functional modules. The numerical coupling module - the core component - serves to treat the feedbacks between the functional modules. A first implemented example that couples an economic and a climate module by means of a two-phase meta-optimization is presented here. The algorithm and mathematical structure behind are discussed as well as important features such as convergence behavior and reliability.

keywords: integrated assessment, modularity, model coupling, climate change

1 Introduction

This paper is about modularization. We focus on modularization as a method of Integrated Assessment (IA) modeling applied to global and climate change problems. Our approach of modularization differs from the

*e-mail: leimbach@pik-potsdam.de

traditional one which for instance is known from ecosystem modeling [1]. Above all, it differs from the current state of the art in IA modeling [2].

IA models combine knowledge from very different disciplines and try to describe the cause-effect relationships between phenomenons from a synoptic perspective [3, 4, 5]. Reduced-form models are used, in general, for each of the components of an IA model. Nevertheless, the range of complexity covered in IA models is rather broad. On the one hand, there are compact models that have only a few pages of code, on the other hand, there are detailed process-oriented models that comprise almost 100 person-years of research and development. Prominent examples are the DICE model [6] and the IMAGE model [7], respectively.

IA studies, applied to global and climate change problems, are driven by varying stakeholder questions. In order to answer these questions, a level of flexibility is needed that can hardly be provided by traditional IA models. The typical IA model is a more or less monolithic model, in which different components, programmed in the same programming language, are strongly interlinked with each other. The software development is tightly managed within a single institution. A counterexample is presented by Messner and Schrattenholzer [8].

Referring to other authors, Janssen [2] summarized some general limitations and drawbacks of IA models. Among other things he identified a lack of attention for methodological issues, which is due to the ad hoc policy-oriented focus of the modelers. Janssen mainly criticized the application of simplified, often linearized optimization models which are limited in representing a number of relevant processes. He mostly argued that advances in IA modeling will result from applying a new type of model (evolutionary models or adaptive complex system models) and new algorithms. We pursue another way of advancing the methodology of IA modeling. We accept that different research groups develop and use different model types. Since we recognize, simultaneously, that attempts to integrate different models are rare [9], we intend to provide a new method of model coupling to ease the exchange of modules.

Based on the concept outlined in [10], a modular approach to IA modeling is presented. Our vision is a set of modules, distributed over different places, that can be combined in a flexible manner in order to construct

answers on different stakeholder questions [1].

We start with a description of the general architecture of the modular approach (section 2) and a brief discussion about decomposition and coupling methods (section 3). For a particular configuration, we present the mathematical structure of the composing modules (section 4) and consider the numerical coupling algorithm in more detail (section 5). Before results of model runs are presented (section 7), we discuss some characteristics of the implementation (section 6). We conclude by adding some remarks on how the tools presented can further be applied.

2 General architecture

We first want to put attention to the distinction between modules and models. Traditionally, functional modules represent parts of models that are not designed to run as self-contained tools. In our approach, however, functional modules are distinguished by their ability to run as stand-alone models. Nevertheless, we call them modules in order to emphasize the modularity of our approach. In this section, the term 'model' is applied only to the integrated framework of modules. We conceptualized a modular approach with the following types of modules being distinguished:

- knowledge domain modules (subdivided into functional modules and data modules),
- job control module,
- numerical coupling modules.

[1] Starting with discussions at PIK and the European Climate Forum [10], followed by similar thinking [11] at the Tyndall Centre (UK), at CIRED (Paris), and other institutions, an initiative called CIAM^n (Community Integrated Assessment Modules) has been set up to carry out modular IA modeling. CIAM^n is a framework within which various configurations of model experiments can be performed with modules produced and implemented on different machines and in different software environments. In particular, modules can be combined across different institutions - a key requirement for further progress in integrated modeling. CIAM^n builds on a range of advanced software tools for coupling heterogeneous modules (see www.european-climate-forum.net/documents.html/, www.pik-potsdam.de/∼linstead/, www.centre-cired.fr/docs/rapportsynth.pdf, www.cs.man.ac.uk/cncsoftiam.pl/). This paper presents results attained with the PIK software system. This includes the first cross-institutional CIAM^n modeling result, since a Tyndall Centre module has been included.

The job control module governs and maintains the communication between the modules. It manages the data transfer between modules and calls them to run. Data are transferred via interface data files and via sockets. This allows not only the coupling of modules programmed in different languages, but also offers platform independence and bears the potential of module coupling via internet. Data modules represent the data bases of several functional modules. They differ from the interface data files. Knowledge domain modules are linked with each other and with the numerical coupling modules by means of interface data files. These files do not represent modules because of their temporary nature. However, once an interface is defined for a module by specifying what the module gets from and puts into the data files, one can replace the current module, which represents an instance of the respective generic module, by another module as long as it meets the requirements of this interface. This provides the key to enhance expansibility and transparency of IA models as discussed by Jaeger et al. [10].

Another key feature of the modular architecture is the autonomy of knowledge domain modules, which therefore can completely be developed, tested and run in a separate way. New challenges, however, arise when we try to couple these modules. In particular, the task is: (1) to cross different programming languages, hardware platforms and remote networks and (2) to capture feedbacks between modules. The focus here is on the latter. We will make a major contribution to tackle this task.

The treatment of feedbacks is fundamental in explaining the differences of the modular and the traditional approach to Integrated Assessment. Whereas in traditional integrated models functional modules interact between or within each time step, the modular approach runs each module separately for the entire time horizon. In order to capture the major feedback effects that still exist within each time step, the single modules have to be run iteratively with some information being exchanged between them.

The numerical coupling module forms the core of the modular approach. It serves to capture feedbacks and balance interactions between the knowledge domain modules. For different kinds of modules to be linked and different kinds of feedbacks to be treated, different numerical coupling modules are required. Nevertheless, there are at least classes of configurations (i.e. problem-specific compositions of modules) that demand for similar numer-

ical coupling modules. We shall put more emphasis on this generalization later on. The first instance of the numerical coupling module was developed to handle normative feedbacks between a simulation module and an intertemporal optimization module.

The implemented examples of the modular approach represent applications that can also be carried out by the traditional approach. We use the optimal solution of the original model (traditionally integrated) as a benchmark. The task is to show that the modular approach will come up with the same result as the original integrated model.

3 Decomposition and coupling methods

Recent attempts to enhance the methodological basis of IA modeling depart from advanced optimization algorithms that simultaneously solve the actual optimization task (e.g. welfare maximization within the economic subsystem) as well as the coupling task (e.g. integration of a climate system). Janssen [2, 12] discussed the following methods: Sequential Reduced-System Programming, Penalty Method [13], Quadratic Programming, and Genetic Algorithms. The issue of efficient algorithms is quite important. However, in contrast to the idea pursued by Janssen, the idea of the modular approach is to handle the external linkages between modules strictly separated from the internal linkages within each module.

The technique of the modular approach to couple modules can be linked to methods known from Mathematical Programming. When dealing with large scale systems or with multiobjective systems [14], decomposition methods were applied. On a higher level, a control entity coordinates the local optimal solutions in an iterative fashion in order to achieve the overall optimum. From the decomposition procedure, a master program [15] or a coordination problem [16] result that bears a resemblance with the numerical coupling module specified below. Similar approaches are also known from hierarchical control and optimization [17]. The Goal Coordination Approach presented by Singh [17] is representative for a broad range of decomposition methods. It splits the original joint optimization problem into single optimization problems based on the duality theorems (i.e. the equivalence of the optimal solutions of the primal and dual problem). While the decentralized systems optimize with given values for the Lagrange multipli-

ers, the coordination system is confronted with the task to determine new values for the Lagrange multipliers based on the solution of the decentralized systems. Singh demonstrated that this Goal Coordination Approach will lead to the optimal solution if there is no duality gap. Similar to the Goal Coordination approach, the approach here represents a gradient-based method, but dissimilar to that, it is not an infeasible method, i.e. the first feasible solution found is not the optimum one.

Most approaches that handle decomposed systems numerically/mathematically modify the objective functions of the subsystems (i.e. knowledge domain modules). We, in contrast, will present an approach based on adding bounds. An equivalence between both is most likely, which has, however, not been tested. Our approach resulted from the intuitional attempt to leave the objective functions of the subsystems, and hence the decision (control) characteristics and the intrinsic dynamics intact.

The application of a numerical coupling algorithm as an autonomous module represents an innovative feature in IA modeling. Bahn et al. [18, 19] reported about such approach within an application close to the IA domain. Bahn et al. were dealing with a multi-region problem where the regions are interconnected by a global target to curb greenhouse gas emissions. While they used the Analytic Center Cutting Plane Method - an interior point method - for numerical solution, the decomposition approach is similar to that of Singh described above. This approach is different from the joint maximization approach [20] which otherwise is used in multi-region modeling. Our numerical coupling module does not serve to coordinate different regions, but to link the dynamics of an economic module and a climate module. While this link is due to a climate guard-rail which is similar to the global emissions reduction target with Bahn et al., the coupling task is different because the delayed reaction of the climate system on the emissions has to be taken into account. This coupling of an optimization and a simulation model is novel.

4 Specification of modules

The problem setting as well as single modules are adopted from the ICLIPS framework (see Toth et al. [21]). There is an economic module (control system) that maximizes intertemporal welfare and a climate module (dynamic

system) that transforms a greenhouse gas emission trajectory, coming from the economic module, into a trajectory of global mean temperature (GMT). Within the business-as-usual evolution, the economic module produces emissions that would lead to a rise of the GMT by more than 4°C above the preindustrial level in the long run. A normative guard-rail is introduced that shall prevent the GMT change to exceed 2°C within the considered time horizon.

According to the traditional approach, the economic system and the climate system would be coupled by representing the dynamics of the climate system as constraints of the intertemporal optimizing framework. Consequently, the guard-rail would be treated as an additional constraint to the dynamics of a joint economy-climate system (cf. [9, 22]). This is quite different within the modular approach. The question arises how the economic module gets the necessary information in order to reduce emissions when both modules are running separately. It is by no means sufficient to just reduce the emissions, since the economic efficiency has to be maintained, while the climate system is kept below the threshold.

The newly developed numerical coupling module, called Meta-Optimizer, will give an answer on the above question. In order to capture feedbacks between the climate and the economic module, that are caused by the climate guard-rail, both domain modules and the Meta-Optimizer are processed iteratively in a sequential manner. Within each iteration, the Meta-Optimizer either activates the phase A or the phase B algorithm. While phase A aims at finding an emission trajectory that keeps the climate guard-rail, phase B allows an expansion of emissions in order to improve welfare. The iterative adjustment process is carried out by means of a barrier, representing an upper bound, on the actual interface variable - the CO_2 emissions.

Before we are going to present the mathematical model of the Meta-Optimizer, we summarize the generic mathematical structure of the economic and climate module. The decision horizon consists of a finite number of periods. Splitting them into increasingly shorter sub-periods approximates continuous time with a finite decision horizon. Thus, although the modules are discrete in nature, we use the continuous form of representing time. Symbols for time derivatives are to be understood in terms of difference equations.

Throughout the model presentation, we use the following indices:

Indices/sets:

t, τ, T, X	indices, sets	time periods
i, I	index, set	regions
r, Q	index, set	iterations.

4.1 Economy module

The major component of the economic module is an economic growth model that includes a welfare (U) maximizing objective function:

$$Max \;\; U^r = \sum_{i=1}^{n} \sum_{t=1}^{m} f[C_i^r(t)] \cdot e^{-\rho t} \tag{1}$$

This welfare function adds up the utilities of the regions' representative households. Utility is a function f of the consumption path *C(t)* subject to discounting by discount rate ρ. For f holds

$$f'[C] > 0; \quad f[C]'' < 0.$$

A production function g with capital K as production factor generates the gross product. We assume

$$g'[K] > 0; \quad g''[K] < 0; \quad g[0] = 0.$$

In each period, the regions have to decide which share of gross product to consume or to invest (control variable I):

$$g[K_i^r(t)] = I_i^r(t) + C_i^r(t) + h[\dot{E}_i^r(t)]. \tag{2}$$

Moreover, some part of the gross product might be spent for climate protection. This expenditure is represented by the mitigation cost h as a function of CO_2 emission change $\dot{E}$ (negative for emission reduction). Emission reduction (representing an additional control variable) is measured as the deviation from a given business-as-usual emission baseline v. For h we assume

$$h'[\dot{E}] < 0, \;\; h''[\dot{E}] > 0 \;\; \text{for } \dot{E} < 0$$

$$h[\dot{E}] = 0 \;\; \text{for } \dot{E}(t) \geq 0.$$

Capital accumulation follows the standard capital stock equation of motion:

$$\dot{K}_i^r(t) = I_i^r(t) - \delta K_i^r(t). \tag{3}$$

with the depreciation rate δ. The associated initial condition assigns the initial value k_i to the capital stock of the first period:

$$K_i^r(0) = k_i. \tag{4}$$

Total CO_2 emission is added up from regional emissions:

$$E^r(t) = \sum_{i=1}^{n} E_i^r(t) \tag{5}$$

and is restricted by the emission barrier $\bar{E}$:

$$0 \leq E^r(t) \leq \bar{E}^r(t). \tag{6}$$

The emission barrier for the first iteration is assumed to be initialized by the business-as-usual emission baseline:

$$\bar{E}^1(t) = v(t). \tag{7}$$

The last two conditions are supplements that serve to support the coupling procedure. For (1)-(7) it holds

$$t \in T, i \in I, r \in Q.$$

4.2 Climate module

The climate module shall be described by equations for the dynamics of the representative climate state vector $\boldsymbol{W}$ and of the atmospheric greenhouse gas concentration B:

$$\dot{\boldsymbol{W}}^r(t) = \psi[\boldsymbol{W}^r(t), B^r(t)] \tag{8}$$

$$\dot{B}^r(t) = \theta[B^r(t), E^r(t), \boldsymbol{F}^r(t)]. \tag{9}$$

The concentration variable is driven by CO_2 emission E, provided by the economic module, and other greenhouse gas emissions $\boldsymbol{F}$. The following initial conditions (with initial climate system vector $\boldsymbol{w}$ and initial greenhouse gas concentration b) hold:

$$\boldsymbol{W}^r(0) = \boldsymbol{w} \tag{10}$$

$$B^r(0) = b. \tag{11}$$

Furthermore, we assume that the dynamics of the other greenhouse gases are given by exogenous scenarios $\boldsymbol{a}(t)$:

$$\boldsymbol{F}^r(t) = \boldsymbol{a}(t). \tag{12}$$

For (8)-(12) it holds

$$t \in T, r \in Q.$$

4.3 Meta-Optimizer

In this subsection, we present only the mathematical structure of the Meta-Optimizer. A detailed discussion follows in the next section.

Additional indices/sets:

Q_1, Q_2	sets	phase A and phase B iterations
q	index	last iteration of phase A before switching to phase B
p	index	last iteration of phase B

Parameters (all positive):

ϵ	real	small number
η	real	convergence criterion
β	real	adjustment parameter
γ	real	adjustment parameter
ω	real	climate guard-rail
l	real	lower bound of emission barrier

Variables:

$\bar{E}$(t)	real	upper boundary of CO_2 emissions (emission barrier)
W(t)	real	GMT change related to preindustrial level
P	real	degree of environmental goal (i.e. guard-rail) violation
R(t)	real	percentage retreat of emission barrier
G	real	goal of phase B (potential welfare increase)
Z(t)	real	increase of emission barrier in phase B
D(t)	real	retreat of emission barrier
μ(t)	real	welfare sensitivity (dual variable) of emission constraint

Functions:

ϕ	function	measurement of environmental goal attainment

4.3.1 Phase A

$$P^r = \sum_{t \in X} [W^r(t) - \omega] \tag{13}$$

$$R^r(t) = [1/(1 + \phi^r(P^r))] + [1 - 1/(1 + \phi^r(P^r))]^{\frac{t}{\Delta t}+1} \tag{14}$$

$$\bar{E}^{r+1}(t) = \max[l, \bar{E}^r(t) \cdot R^r(t)] \tag{15}$$

$$D^r(t) = \bar{E}^{r+1}(t) - \bar{E}^r(t) \tag{16}$$

For (11)-(16) it holds:

$$t \in T, r \in Q_1, Q_1 \subset Q, X \subseteq T, X = \{t | W^r(t) > \omega\}.$$

4.3.2 Phase B

$$Max \;\; G^r = \sum_{t=1}^{m} Z^r(t) \cdot (\mu^r(t) - \epsilon) \tag{17}$$

s.t.

$$\sum_{t=1}^{m} Z^r(t) \leq \beta \cdot \sum_{t=1}^{m} D^q(t) \tag{18}$$

$$|Z^r(t)| \leq \gamma \cdot \max_{\tau}[D^q(\tau)] \tag{19}$$

$$\bar{E}^{r+1}(t) = \bar{E}^r(t) + Z^r(t) \tag{20}$$

$$Z^r(t) \geq 0 \tag{21}$$

Terminal Condition

$$\sum_{t=1}^{m} |E^{p+1}(t) - E^p(t)| < \eta \tag{22}$$

For (17)-(22) it holds:

$$t \in T, \tau \in T, r \in Q_2, Q_2 \subset Q.$$

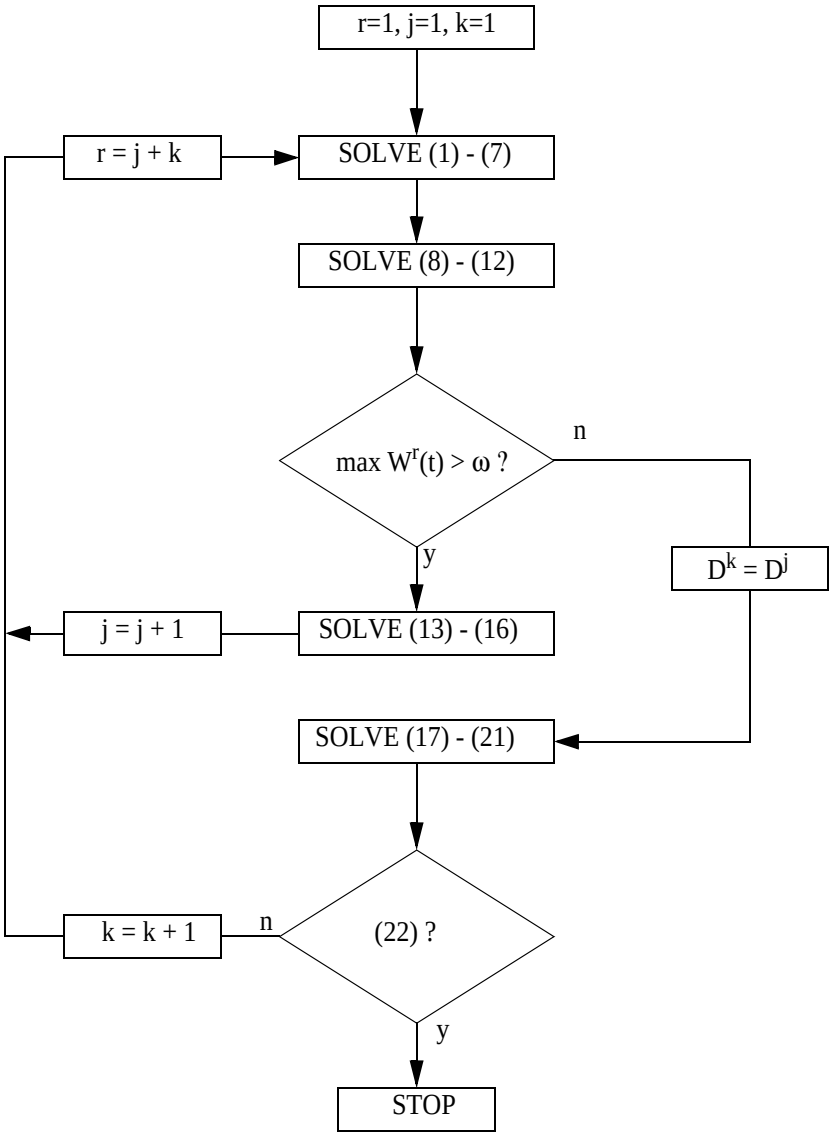

Figure 1: Meta-optimizing algorithm

5 Iteration algorithm

Based on the indexation and numeration of the previous section, Figure 1 illustrates the iteration algorithm. It can be summarized as follows:

1. set initial emission barrier
2. retreat emission barrier as long as climate guard-rail is violated
3. relax emission barrier in order to improve welfare
4. if climate guard-rail is violated again, go back to 2
5. test convergence and stop.

The newly introduced emission barrier $\bar{E}(t)$ that constraints emission dynamics within the economic module (eq. 6), and its shadow prices μ(t) represent the most essential elements of the coupling procedure. Within each iteration, the Meta-Optimizer determines a new emission barrier (eq. 15 and eq. 20, respectively) by either activating the phase A algorithm or the phase B algorithm.

Phase A (retreat phase) aims at finding an emission trajectory that keeps the climate goal. The emission barrier is gradually lowered, assuming that $\partial W_{max}/\partial E \geq 0$ holds (with W_{max} representing the maximum of GMT change). The functional form of the applied retreat function (eq. 14) serves to determine an appropriate decline of the emission barrier which in turn depends on the deviation of the actual temperature change trajectory from the climate guard-rail (eq. 13). Since the GMT change in each point of time does not only depend on the emission in this time period but to a greater extent on the entire historical emission path, it is not appropriate to adjust the emission barrier synchronously in time with the deviation measure. Instead, deviations are accumulated in variable P. Over different domains of P and different co-domains of ϕ, respectively, the process of barrier adjustment can be carried out flexibly (see Figure 2).

We devised the retreat function heuristically. Its functional form is not decisive for finding the optimal solution (see below), but it contributes to reduce the number of iterations to find this optimum. In this application, we further specify:

$$\phi(P) = 15\omega + \begin{cases} 0 & : \quad P > 25 \\ 300 - 10P & : \quad P \leq 25, \end{cases}$$

that means

$$\phi' \leq 0$$

and

$$15\omega \leq \phi \leq 300 + 15\omega.$$

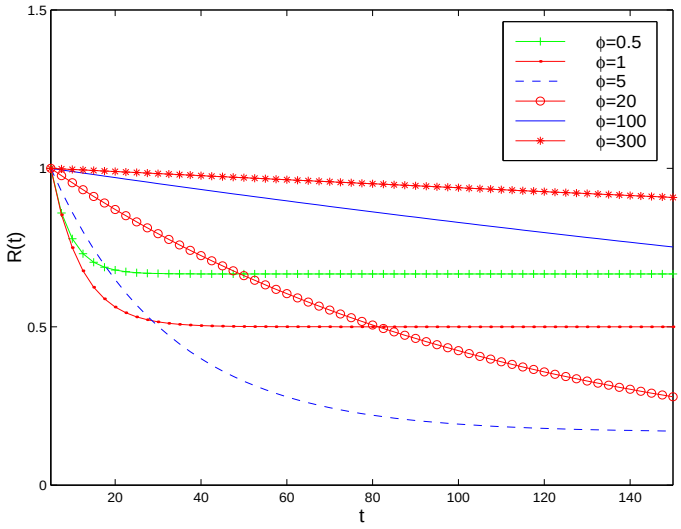

Figure 2: Retreat function

Assuming ω=2, ϕ is between 30 and 330. Figure 3 shows the impact of these ϕ-values on the retreat of a baseline barrier (with Δt=5). In general, if the level of climate goal violation P is rather big (and ϕ is close to its lower boundary), the emission barrier is lowered significantly. This specification, however, excludes small values for ϕ which actually cause significant retreats already in the short term (see Figure 3 for ϕ=1), and which therefore might be more effective in order to meet the climate goal. The rationale here is that the economy is less constrained when emissions are more restricted in the long term. This specification leads to less iterations in phase B as well as in total. Consequently, within each iteration, the barrier retreat in later model periods is higher than in initial periods. And if P decreases, retreats will decrease too.

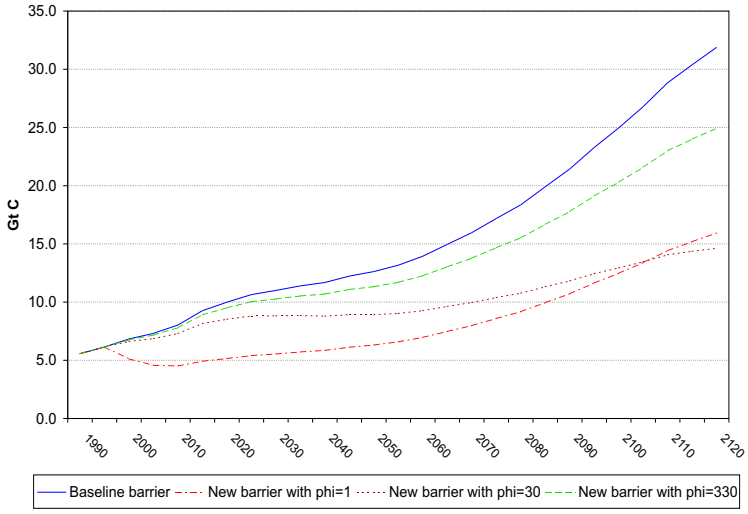

Figure 3: Retreat of emission barrier

Phase A ends when an emission trajectory is generated by the economic module which corresponds with a GMT trajectory that fulfills the climate constraint. Due to the rather non-linear interaction of the GMT trajectory and the evolution of emissions, a multitude of emission trajectories can meet the climate constraint. This gives some space for improving welfare. Phase B (relaxation phase) aims at finding the emission trajectory that keeps the climate constraint with highest economic welfare (the original objective of the economic module). Therefore, the algorithm relaxes the emission barrier in periods with high welfare sensitivities. The objective function in phase B (eq. 17) maximizes the product of the barrier increase and the shadow prices of the respective barrier points (as determined by the economic module) cumulated over the entire time horizon. The change of the emission barrier in each time period as well as the cumulated amount of barrier relaxations is limited (eq. 18 and eq. 19) to a certain share of the single maximum decrease and the cumulated decrease of the emission barrier, as determined by the last iteration in phase A (eq. 16). Convergence is tested in equation (22).

We now demonstrate that in the absence of multiple optima the presented iteration algorithm will yield an optimal solution which corresponds to the solution of the analogous joint economy-climate model (the joint

model treats the dynamics of the climate system as a constraint of the intertemporal optimization). Let us introduce U^* as the optimal solution of the joint optimization problem, $\tilde{U}$ as the optimal solution of the unconstrained economic model (1)-(5), and $\hat{U}$ as the optimal solution of the constrained economic system (1)-(7) with $\bar{E}$ being the result of (8)-(12) and (13)-(22). The following relation holds:

$$\hat{U} \leq U^* \leq \tilde{U}.$$

First, the phase A algorithm serves to bring the system into the feasible solution space ($W(t) \leq \omega$). Then, given the convex model structure of the economic module, in phase B it holds (remember index p representing the last iteration in phase B):

$$\bar{E}^{r+1}(t) \geq \bar{E}^r(t) \ \forall \ t \implies U^{r+1} \geq U^r,$$

and if

$$\exists t(\mu^r(t) > 0) \implies U^{r+1} > U^r.$$

Hence, if

$$\exists t(\mu^r(t) > 0) \ \forall (r < p)$$

then for

$$r \to p \quad \implies \quad \begin{cases} (1) & : & U^r \to U^* \\ (2) & : & \mu^r(t) = 0 \ \forall \ t \\ (3) & : & \exists t(W^r(t) > \omega). \end{cases}$$

In case (1) the corresponding optimal solution is found. This also applies to case (2), because if all dual variables μ equal to zero, than, due to the Kuhn-Tucker conditions [25, 26], the emission boundary (eq. 6) is no longer binding and the achieved optimal solution is equivalent to that of the unconstrained economic model. Taking the above relationship between U^*, $\hat{U}$, and $\tilde{U}$ into account, we get

$$\hat{U} = \tilde{U} \implies \hat{U} = U^*.$$

In case (3) the algorithm switches back to phase A to search for a new feasible solution. If it can be guaranteed that each re-entry into phase B

comes along with a higher welfare, U^* will be reached iteratively [2]. Assuming infinitesimal changes of $\bar{E}(t)$, this is the case here. While each relaxation step optimally increases welfare, each retreat step of the same magnitude reduces the welfare, no matter what retreat function is applied, by a lower amount or the same amount at maximum. Convergence is assured since the latter can be excluded for $\beta < 1$ in constraint (18).

If the assumption

$$\exists t(\mu^r(t) > 0)$$

is not met, then again, since μ cannot become negative, the conclusion of case (2) will apply.

Infinitesimal changes of the emission barrier $\bar{E}(t)$ will result from $\beta << 1$. In practice, however, β as well as γ in constraints (18) and (19) will be chosen in a way that changes $\bar{E}(t)$ to discrete amounts. This helps to converge to the optimal solution in reasonable time. But if constraint (18) is not met as an equation, a positive value could be assigned to some $\mathrm{Z}^r(\tau)$, although $\mu(\tau)$=0. Such a relaxation in periods without sensitivity could prevent convergence. With subtracting epsilon (ϵ >0) from the shadow prices in equation (17), this, however, cannot happen, since this would have a negative impact on the objective function.

Several modifications of the algorithm have been tested:

- update D in phase B
- replace max[D^q(t)] in (19) by D^q(t)
- allow Z(t) to become negative.

With respect to the convergence behavior, all of them are dominated by the specification presented in section 4. The algorithm is, however, still not optimized with respect to the speed of convergence. A fine-tuning of the parameters (and possibly an adjustment of function ϕ) will become necessary, if the climate guard-rail is varied and if different instances of the generic domain modules are applied. Nevertheless, as we shall demonstrate later, the same initialization of parameters works for different settings. The

[2]Note that convergence towards U^* cannot be guaranteed, if SO_2 emissions become an additional interface variable controlled by the economic actors. Due to the opposite warming effect of CO_2 and SO_2, multiple optima may occur. This is especially critical, if guard-rails in the form of temperature change rates are considered.

characteristic feature of the convergence behavior of the algorithm is the fact that a good approximation of the final solution can be achieved after a few repeated phase A and phase B iterations. A more accurate solution will result from reducing parameter η. This, however, is at the cost of a progressively increasing number of iterations and therefore computing time.

6 Implementation

The ICLIPS climate model (see [23]) and aggregated economic model - called ICEMODE (see [24]) - represent the instances of the climate and economic module [3]. Changes of the original mathematical structure of both models are restricted to the insertion of an emission barrier into the economic model. Figure 4 shows the ensemble of modules (here called CIAM^n configuration[4]) and the respective interface data files. The three interface data files include an emission barrier trajectory (contr.dat), an emission trajectory (totemiss.dat), and a GMT change trajectory (temp.dat). The file 'totemiss.dat' contains also the shadow price information of the emission barrier constraint and the value of the welfare function. The instance of the job control module, called CIAM^n driver, operates in the background. The driver takes care that the modules start when input data have changed. The Meta-Optimizer (for the GAMS program code see Appendix) checks whether the solutions converge and terminal conditions are met. If so, then the CIAM^n driver stops the computation. Otherwise, the Meta-Optimizer determines a new emission barrier.

While the number of iterations mainly depends on the parameters β and γ, and the tolerance criterion η, the computation time depends on the complexity of the knowledge domain modules, the numerical algorithms and the hardware platform. The multiregional economic module is the most time-consuming module in our setting due to its intertemporal optimization approach. Within each CIAM^n iteration, thousands of numerical iterations are needed to solve the economic optimization module. All modules are programmed in GAMS and use the incorporated algorithms for solving non-

[3] While the climate model runs until 2200 (in five-year time steps), ICEMODE determines the economic dynamics (investment, consumption) and the CO_2 emissions until 2110 only. Emissions are kept constant thereafter.

[4] CIAM^n is the abbreviation for Community Integrated Assessment Modules.

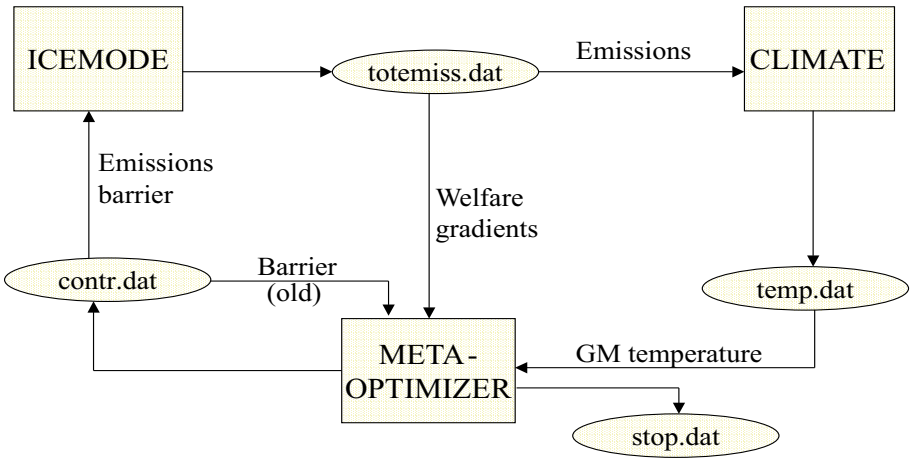

Figure 4: CIAM^n structure

linear programming problems [27, 28] (see also www.gams.com), besides the CIAM^n driver which is coded in PYTHON. Running all modules on a workstation (IBM RS6000), each CIAM^n iteration takes around 12 minutes. Computing time is reduced to less than 2 minutes for each iteration, when running the economy module on a super parallel computer.

Before looking at some results, we make an additional step forward. We made use of the modular architecture which allows the replacement of a domain module without interfering with the code of the other modules. We replaced the ICLIPS climate module by the MAGICC model [29, 30, 31]. We integrated a version of MAGICC which was applied for studies of the IPCC as documented in the Third Assessment Report on Climate Change [32]. MAGICC is programmed in FORTRAN. It is not our intention to discuss the differences of model results caused by different climate modules in detail. We just want to demonstrate the practicability of the underlying key feature of modularity.

7 Results

Analyzing the results, our major focus is on convergence and reliability. As to the latter, we consider results from the original ICLIPS model which hard-wires the economic and climate module, as benchmark. This, of course, only applies to CIAM^n runs with the ICLIPS climate module being included.

Furthermore, the comparison of the two instances of this CIAM^n configuration, which are distinguished by the climate module employed, will allow for drawing some conclusions with respect to reliability. We ran the model within the given settings for two different normative climate guard-rails:

1. $\omega = 2.0$
2. $\omega = 3.0$.

Figure 5 and 6 demonstrate convergence for all model runs after a different number of iterations. Convergence is measured in terms of welfare and in terms of the maximum of the temperature change time series which actually should be constrained to the normatively given guard-rail. There is a similar convergence pattern for both CIAM^n variants. Not surprisingly, convergence takes the more time the more significant the divergence between the climate guard-rail and the business-as-usual evolution of the climate system is. With a guard-rail of 2.0°C, it needs around 9 and 12 CIAM^n iterations, respectively, to bring the economic system into a state where it, for the first time, produces emissions that allow the climate system to stay below the temperature threshold over the entire time horizon. A few CIAM^n iterations follow that, due to relaxation of the emission barrier, result in higher welfare, but again drive the climate system out of the permitted space. The zigzag pattern indicates frequent switches between phase A and phase B. However, it takes always just one CIAM^n iteration to bring the system back into the feasible space. But increasing the welfare becomes harder and harder. This is best documented in Figure 6 where it can be seen that between the 13th and 30th iteration, the welfare increments are most significant.

With a guard-rail of 3°C, the feasible climate space is reached already after less than 10 iterations. Convergence can be documented after almost 35 iterations and few switches between phase A and phase B. We prolonged the iteration cycle by narrowing the tolerance criterion η. The relaxation phases become longer, the differences between the temperature change maximums become smaller (see Figure 5). Moreover, there is hardly any increment of welfare at the end of the consecutive phases B (see Figure 6). In addition to the long relaxation phases, the variant with the least strict climate goal is distinguished by the deep fall back of the maximum of the temperature

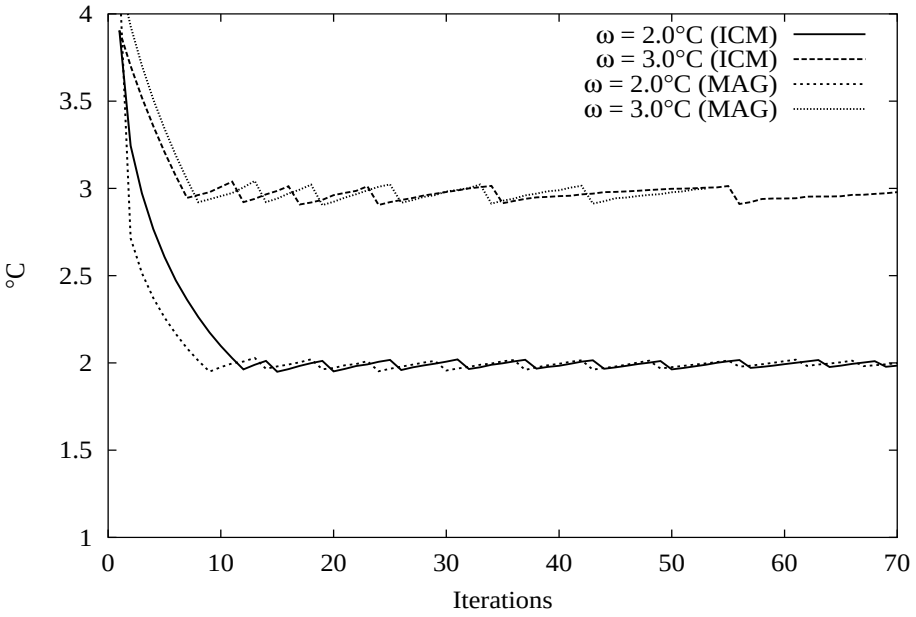

Figure 5: Convergence of GMT change maximum (ICM: CIAM^n run with the ICLIPS climate module; MAG: CIAM^n run with the MAGICC module)

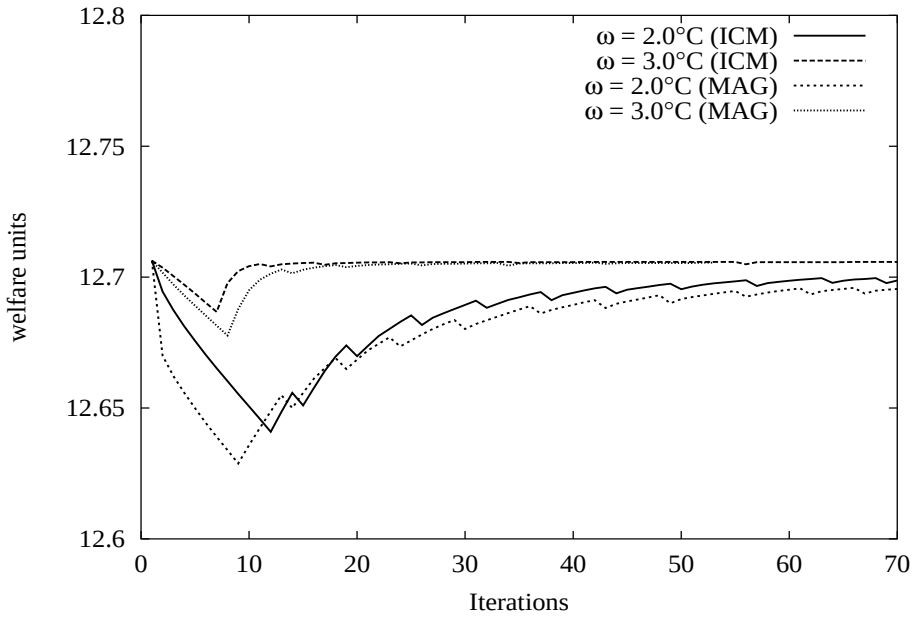

Figure 6: Convergence of welfare measure (ICM: CIAM^n run with the ICLIPS climate module; MAG: CIAM^n run with the MAGICC module)

change trajectory when switching from phase A to phase B. This is due to the generally higher level of the emission barrier which, subject to a similar percentage retreat factor, yields a more substantial decline. Nevertheless, the results demonstrate that the coupling algorithm can be applied to different guard-rails. Further experiments have shown that this even applies to settings with guard-rails that vary over time.

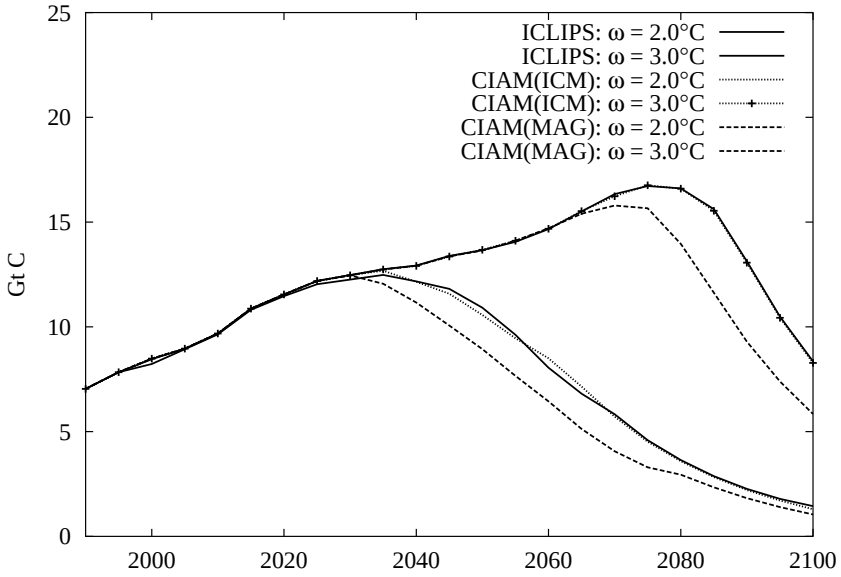

Figure 7: Global CO_2 emissions

The results that we obtained from the CIAM^n runs with the ICLIPS climate module are remarkably close to those of the original coupled climate-economy model run. The welfare values are 12.7055 and 12.7002, respectively, for the ICLIPS model and 12.7058 and 12.6996, respectively, for the CIAM^n model. Moreover, Figure 7 demonstrates a correspondence in the resulting emission trajectories. Such a correspondence cannot be expected from CIAM^n runs with the MAGICC climate module. Due to differences in the internal dynamics of the climate modules (resulting in a higher climate sensitivity of MAGICC), the respective optimal emission trajectories deviate from each other. Nevertheless, the same convergence pattern (as documented in Figure 5 and Figure 6) indicates the robustness of the coupling algorithm for different instances of the climate domain module.

8 Conclusions and perspectives

We presented a new approach to model coupling that consequently pursued the idea of modularity. To our knowledge, the first application of this pioneering concept for the coupled climate-economy system is presented in this paper. Together with the conceptional outline in [10], this approach opens a new and more efficient way of international division of labor in carrying out Integrated Assessment studies.

Besides of the technical details, the following core elements of the modular approach can be summarized:

- a job control module which governs the communication between different modules and guarantees independency from programming languages, operating systems and hardware platforms
- a numerical coupling module which is able to treat the feedbacks between autonomous functional modules.

Our first experiments show encouraging results for a particular configuration. The reliability of our approach was demonstrated by the close correspondence of the results from alternative module settings and with those of the traditional approach. Some more general conclusions can also be drawn. The iterative algorithm of module coupling that we developed is characterized by two features that can likely be applied to other problems and configurations as well. First, this relates to the separation into two different phases which help to reconcile different objectives (climate goal and economic goal) that can be ordered lexically. Second, this applies to the introduction of a barrier for the actual interface variable between the functional modules. Adjusting this barrier seems to be a promising way of capturing feedbacks. The mechanics of our algorithm is mainly based on this adjustment process.

While at a first glance the modular approach seems to be restricted only by the computational power, other limits exist and have to be kept in mind. Non-convex module structures will challenge the robustness of the coupling algorithm. Furthermore, each module represents a model which has its own boundary conditions. Due to meta-optimization and data exchange, there is a permanent change in the boundary conditions that defy the control of the

modeler or model user. The change of boundary conditions may cause the models to fail. The modular approach can therefore only be controlled, if there is a limited number of interface variables between the modules. Highly interconnected modules have to be treated as a single module, hence must be developed and computed simultaneously.

While the combination and easy replacement of modules that will be developed by different research groups autonomously is the objective of the modular approach, one cannot expect an universal coupling algorithm that is applicable to any kind of configuration. Future research has to demonstrate the practicability of the modular approach for different configurations. This demands for the investigation of module settings that include combined feedbacks (the present approach only treats a normative feedback), of module settings with multiple interface variables, and of problems with single and multiple interface variables within configurations of more than two knowledge domain modules.

Acknowledgement We would like to thank Tom Wigley and the Tyndall Centre, University of East Anglia, for providing us with the code of the MAGICC model. MAGICC was written by Tom Wigley and Sarah Raper, then at the University of East Anglia. Sarah Raper is still at the University of East Anglia whilst Tom Wigley is now at the National Center for Atmospheric Research in Boulder. We would also like to thank Cezar Ionescu (Potsdam Institute for Climate Impact Research) for fruitful discussions and valuable aid in getting the modular framework operational.

References

[1] J.F. Reynolds and B. Acock, Modularity and genericness in plant and ecosystem models, Ecological Modelling 94(1997) 7-16.

[2] M.J. Janssen, Modelling Clobal Change: The Art of Integrated Asessment Modelling, Edward Elgar, Celtenham, 1998.

[3] J. Rotmans and H. Dowlatabadi, Integrated Assessment Modeling, in: Human Choice and Climate Change Tools for Policy Analysis, eds. S. Rayner and E. Malone, Batelle Press, Columbus, 1998, pp. 291-377.

[4] J. Rotmans and M.B.A. van Asselt, Integrated Assessment: A Growing Child on its Way to Maturity, Climatic Change 34(1996) 327-336.

[5] S.H. Schneider, Integrated Assessment Modeling of Global Climate Change: Transparent Rational Tool for Policymaking or Opaque Screen Hiding Value-Laden Assumptions?, Environmental Modeling and Assessment 2(1997) 229-248.

[6] W.D. Nordhaus, Managing the global commons, MIT Press, Cambridge, 1994.

[7] J. Alcamo, IMAGE 2.0 - Integrated Modeling of Global Climate Change, Kluwer, Dordrecht, 1994.

[8] S. Messner and L. Schrattenholzer, MESSAGE-MACRO: linking an energy supply model with a macroeconomic module and solving it iteratively, Energy 25(2000) 267-282.

[9] P.S. Gaertner, Optimisation analysis and integrated models of the enhanced greenhouse effect, Environmental Modelling and Assessment 6(2001) 7-34.

[10] C. Jaeger, M. Leimbach, C. Carraro, K. Hasselmann, J.C. Hourcade, A. Keeler and R. Klein, Community Integrated Assessment: Modules for Cooperation, Working Paper, FEEM Nota di Lavoro 53.2002, Fondazione Eni Enrico Mattei, Milan, 2002.

[11] H.J. Schellnhuber, R. Warren, A. Haxeltine and L. Naylor (2002), Developments in Integrated Assessment: The Co-productive Approach, Proceedings of OECD Workshop on Benefits of Climate Policy, December 2002, Paris, in press.

[12] M.J. Janssen, Meeting Targets: Tools to Support Integrated Assessment Modelling of Global Change, CIP-Gegevens Koninklijke Bibliotheek, Den Haag, 1996.

[13] D.G. Luenberger, Optimization by Vector Space Methods, John Wiley, New York, 1969.

[14] R. Caballero, T. Gomez, M. Luque, F. Miguel and F. Ruiz, Hierarchical generation of Pareto optimal solutions in large-scale multiobjective systems. Computers & Operations Research 29(2002) 1537-1558.

[15] G.B. Dantzig and P. Wolfe, Decomposition principle for linear programs, Operations Research 8(1960) 101-111.

[16] J.F. Shapiro, Mathematical Programming: Structures and Algorithms, John Wiley, New York, 1979.

[17] M.G. Singh, Dynamical Hierarchical Control, North-Holland, Amsterdam, 1980.

[18] O. Bahn, A. Haurie, S. Kypreos and J.-P. Vial, Advanced mathematical programming modeling to assess the benefits from international CO_2 abatement cooperation, Environmenmtal Modeling and Assessment 3(1998) 107-115.

[19] O. Bahn, A. Haurie, S. Kypreos and J.-P. Vial, A decomposition approach to multiregional environmental planning: A numerical study, in: Operations research and Environmental Management, eds. C. Carraro and A. Haurie, Kluwer, Dordrecht, 1996, pp. 119-132.

[20] P.B. Dixon, The Theory of Joint Maximization, North-Holland, Amsterdam, 1975.

[21] F.L. Toth, T. Bruckner, H.-M. Fuessel, M. Leimbach and G. Petschel-Held, Integrated Asessment of Long-term Climate Policies: Part 1 - Model Presentation, Climatic Change, 56(2003) 37-56.

[22] M. Leimbach and T. Bruckner, Influence of Economic Constraints on the Shape of Emission Corridors, Computational Economics 18(2001) 173-191.

[23] T. Bruckner, G. Hooss, H.-M. Fuessel and K. Hasselmann, Climate System Modeling in the Framework of the Tolerable Windows Approach: The ICLIPS Climate Model, Climatic Change 56(2003) 119-137.

[24] M. Leimbach and F.L. Toth, Economic Development and Emission Control over the Long Term: The ICLIPS Aggregated Economic Model, Climatic Change 56(2003) 139-165.

[25] A. Stolbjerg Drud and Leon Lasdon, Nonlinear Programming, in: Advances in Sensitivity Analysis and Parametric Programming, eds. T. Gal and H.J. Greenberg, Kluwer, Boston, 1997, p. 10-4.

[26] D.G. Luenberger, Introduction to Linear and Non-linear Programming, Addison-Wesley, Menlo Park, 1984.

[27] A. Brooke, D. Kendrick and A. Meeraus, GAMS - A User's Guide, Release 2.25, The Scientific Press, San Francisco, 1992.

[28] A. Stolbjerg Drud, CONOPT - A Large-Scale GRG Code, ORSA Journal on Computing 6(1994) 207-216.

[29] T.M.L. Wigley and S.C.B. Raper, Thermal expansion of sea water associated with global warming, Nature 330(1987) 127-131.

[30] T.M.L. Wigley and S.C.B. Raper, Implication for climate and sea level of revised IPCC emissions scenarios, Nature 357(1992) 293-300.

[31] T.M.L. Wigley, S.C.B. Raper, S. Smith and M. Hulme, The MAGICC/SCENGEN Climate Scenario Generator: Version 2.4: Technical Manual, Climatic Research Unit, UEA, Norwich, 2000.

[32] IPCC, Climate Change 2001: The Scientific Basis, Contribution of Working Group I to the Third Assessment Report of the Intergovernmental Panel on Climate Change, Cambridge University Press, 2001.

A GAMS program

```
$TITLE Meta-Optimizer for economic-climate system

*************
 SETS
*************
  t   time
        /1990,1995,2000,2005,2010,2015,2020,2025,2030,2035,2040,2045,2050,2055,
        2060,2065,2070,2075,2080,2085,2090,2095,2100,2105,2110,2115,2120,2125,2130,
        2135,2140,2145,2150,2155,2160,2165,2170,2175,2180,2185,2190,2195,2200/
  jr  atmospheric gases released by industrial processes /co2,so2/
;

******************
* includefiles
******************

$include d:\piam\control\temp.dat
$include d:\piam\control\baseline.dat
$include d:\piam\control\totemiss.dat
$include d:\piam\control\zf.dat
$include d:\piam\control\contr.dat

* temp.dat provides the temperature time series T2M(t)
* baseline.dat provides the global emission baseline EMT(t,jr) - only first iteration
* totemiss.dat provides the global emission EMT(t,jr) - all but the first iteration
*               and the welfare gradients of emissions barrier grd(t)
* zf.dat provides the current welfare value zf
* contr.dat provides the barrier values d(t), db(t), zb(t), the barrier change value
*               dta(t), and the auxiliary parameter aux

******************
 SCALARS
******************
        epsilon         tolerance parameter /0.001/
        eta             convergence parameter /0.2/
        elo             minimum of total emission /0.1/
        beta            adjustment parameter   /0.4/
        gamma           adjustment parameter   /0.4/
        tempmax         normatively set temperature guard-rail /2.0/
;

******************
 PARAMETERS
******************
          P             environmental utility
          Phi           argument of retreat function
          diff(t)       distance between actual temperature and temperature guard-rail
          R(t)          reduction operator
          emuplo(t)     lower bound of emission barrier (ensure feasibility)
          maxdta        maximum recent barrier change
          maxdiff       maximum distance from guard-rail
          maxtemp       maximum of temperature trajectory
          ZDA(t)        emission barrier of previous iteration
          ZD(t)         current emission barrier
          ZB(t)         benchmark emission trajectory from previuos phase B
          dev           change of emission barrier
          bin           stop parameter
;

***********************************
* Initialisation/Computing of  P
***********************************
```

```
  maxtemp=0;
  P=0;
  dev=0;

  emuplo("1990")=7.0411;
  emuplo("1995")=7.8420;

* the lower emission bound is derived from an annual emission reduction of 4%
  LOOP(t $(ord(t)>2), emuplo(t)=0.86*emuplo(t-1));

* first "old" barrier is initialzed by the value of the emission baseline (second
* part of the addend)
  ZDA(t)=d(t)+BEMT(t,"co2")$(d(t)=0);

  diff(t)=T2M(t)-tempmax;
  loop(t,
        if (diff(t)>0,
                P=P+diff(t);
        else
                ZD(t)=ZDA(t);
        );
  );

* So far P represents the cumulated positive deviation of the current temperature
* trajectory from the guard-rail. Now it is heuristically transformed into a
* nondimensional environmental utility measure. It is further adjusted depended
* on the guard-rail level in order to become a sensitive argument of the retreat
* function below.

  P=15*tempmax+max(0,300-15*P);
  Phi = 15*tempmax + P;

  maxdiff=diff("1990");
  Loop (t,maxdiff=max(diff(t),maxdiff));

*+++++++++++++++++++++++++++++++++
*. Phase A (Retreat)
*+++++++++++++++++++++++++++++++++

  IF (maxdiff > epsilon,

* here comes the retreat function
        R(t)=(1/(1+Phi))+(1-1/(Phi+1))**ord(t);
        display R;

* compute a new emission barrier
        ZD("1990")=ZDA("1990");
        LOOP(t, ZD(t)=R(t)*ZDA(t);
            IF (zd(t)<elo, zd(t)=elo);
            IF (zd(t)<emuplo(t), zd(t)=emuplo(t));
        );

* store last barrier from phase B
        if (aux=1,
                db(t)=zb(t);
        );

        aux=0;
        bin=1;
        dta(t)=zda(t)-zd(t);
        display zd,dta;
  );

  maxdta=0;
  Loop (t,maxdta=max(dta(t),maxdta));
```

```
*****************************
* Meta-Optimization/Phase B
*****************************

VARIABLES
        GF
        zt     hypothetical increase of the emission threshold
;

Positive Variable zt;

        zt.l(t)=1;
        zt.fx("1990")=0;
        zt.fx("1995")=0;

EQUATIONS
        objective
        ZTsum
        ZTsing(t)
;

  objective ..  GF =e= sum(t, zt(t)*(grd(t)-epsilon));

  ZTsum.. sum(t, zt(t)) =L= beta * sum(t, dta(t));

  ZTsing(t) .. abs(zt(t)) =L= gamma * maxdta;

  MODEL CONTROL /ALL/;

  OPTION iterlim=500;
  OPTION reslim=2000;
  OPTION sysout=off;
  OPTION solprint=on;

  SOLVE CONTROL USING dnlp MAXIMIZING GF;

***********************
* Phase B/Evaluation
***********************

  IF (maxdiff le epsilon,

* compute new emission barrier
      ZD(t)=ZDA(t)+zt.l(t);
      bin=1;
      LOOP(t,
            IF (zd(t)<elo, zd(t)=elo);
            IF (zd(t)<emuplo(t), zd(t)=emuplo(t));
      );

* compute convergence measure dev and update zb by current emission trajectory
      IF ((abs(maxdiff)<epsilon),
                dev=sum(t,abs(emt(t,"co2")-db(t)));
           zb(t)=emt(t,"co2");
           aux=1;
      );

***********************
* Termination condition
***********************
      IF ((aux=1) and (dev < eta) and (abs(maxdiff)<epsilon),
                 bin=0;
      );
  );
```